EXAMINER'S AMENDMENT

1. An examiner's amendment to the record appears below. Should the changes and/or additions be unacceptable to applicant, an amendment may be filed as provided by 37 CFR 1.312. To ensure consideration of such an amendment, it MUST be submitted no later than the payment of the issue fee.

2. Authorization for this examiner's amendment was given in a telephone interview with Soyeon (Karen) Pak Laub (Reg No. 39,266) on May 1, 2009.

3. The claims have been amended as follows:

a. Claim 1: replace claim 1 with the following:

1. A system for executing computing tasks in a preboot execution environment, comprising:

a computer readable medium comprising:

a language agent with a preboot execution language interpreter, the language agent from a second system; and

at least one specification for performing at least one computing task in the preboot execution environment, the at least one specification from the second system,

wherein the language agent is configured to interpret the at least one specification for performing at least one computing task in the preboot execution environment, and configured to perform the at least one computing task specified,

wherein the at least one specification from the second system is an encapsulation, encapsulating parameters to be resolved at the system at execution time by the preboot execution language interpreter from the second system,

wherein the language agent is configured to generate one or more encapsulations and to interpret one or more encapsulations,

wherein the language agent is based on an encapsulated object-oriented polyphase language, and

wherein while executing an encapsulation, the language agent is configured to generate another encapsulation.

b. Claim 6: replace claim 6 with the following:

6. A system for image installation in a preboot execution environment, comprising:

a computer readable medium comprising:

a language agent with a preboot execution language interpreter, the language agent from a second system; and

at least one specification for performing at least one task for image installation in the preboot execution environment, the at least one specification from the second system,

wherein the language agent is configured to interpret the at least one specification, and configured to perform the at least one task specified,

wherein the at least one specification is an encapsulation, encapsulating parameters to be resolved at the system at execution time by the preboot execution language interpreter from the second system,

wherein the language agent is configured to generate one or more encapsulations and to interpret one or more encapsulations,

wherein the language agent is based on an encapsulated object-oriented polyphase language, and

wherein while executing an encapsulation, the language agent is configured to generate another encapsulation.

c. Claim 12: replace claim 12 with the following:

12. A system for remote imaging in a preboot execution environment, comprising:

a computer readable medium comprising:

a language agent with a preboot execution language interpreter, the language agent from a second system; and

at least one specification for performing at least one task for remote imaging over a network in the preboot execution environment, the at least one specification from the second system,

wherein the language agent is configured to interpret the at least one specification, and configured to perform the at least one task specified,

wherein the at least one specification is an encapsulation, encapsulating parameters to be resolved at the system at execution time by the preboot execution language interpreter from the second system,

wherein the language agent is configured to generate one or more encapsulations and to interpret one or more encapsulations,

wherein the language agent is based on an encapsulated object-oriented polyphase language, and

wherein while executing an encapsulation, the language agent is configured to generate another encapsulation.

d. Claim 18: replace claim 18 with the following:

18. A system for remote booting over a network, comprising:

a computer readable medium comprising:

a language agent with a preboot execution language interpreter, the language agent from a second system; and

at least one specification for performing at least one task in a preboot execution environment for remotely booting a computer over a network, the at least one specification from the second system,

wherein the language agent is configured to interpret the at least one specification, and configured to perform the at least one task specified,

wherein the at least one specification is an encapsulation, encapsulating parameters to be resolved at the system at execution time by the preboot execution language interpreter from the second system,

wherein the language agent is configured to generate one or more encapsulations and to interpret one or more encapsulations,

wherein the language agent is based on an encapsulated object-oriented polyphase language, and

wherein while executing an encapsulation, the language agent is configured to generate another encapsulation.

e. Claim 24: replace claim 24 with the following:

24. A method for executing computing tasks in a preboot execution environment, comprising:

receiving by a first system from a second system a language agent with a preboot execution language interpreter;

generating one or more encapsulations by the language agent;

interpreting one or more encapsulations by the language agent;

receiving by the first system from the second system at least one specification for performing at least one computing task in the preboot execution environment;

interpreting by the language agent the at least one specification for performing at least one computing task in the preboot execution environment;

performing the at least one computing task specified; and

while executing an encapsulation, generating another encapsulation,

wherein the at least one specification is an encapsulation, encapsulating parameters resolved at the first system at execution time by the preboot execution language interpreter from the second system, and

wherein the language agent is based on an encapsulated object-oriented polyphase language.

f. Claim 28: replace claim 28 with the following:

28. A method for image installation in a preboot execution environment, comprising:

receiving by a first system from a second system a language agent with a preboot execution language interpreter;

generating one or more encapsulations by the language agent;

interpreting one or more encapsulations by the language agent;

receiving by the first system from the second system at least one specification for performing at least one task for image installation in the preboot execution environment;

interpreting by the language agent the at least one specification for performing at least one task for image installation in the preboot execution environment;

performing the at least one task for image installation specified; and

while executing an encapsulation, generating another encapsulation,

wherein the at least one specification is an encapsulation, encapsulating parameters resolved at the first system at execution time by the preboot execution language interpreter from the second system, and

wherein the language agent is based on an encapsulated object-oriented polyphase language.

g. Claim 33: replace claim 33 with the following:

33. A method for remote imaging in a preboot execution environment, comprising:

receiving by a first system from a second system a language agent with a preboot execution language interpreter;

generating one or more encapsulations by the language agent;

interpreting one or more encapsulations by the language agent;

receiving by the first system from the second system at least one specification for performing at least one task for remote imaging in the preboot execution environment;

interpreting by the language agent the at least one specification for performing at least one task for remote imaging in the preboot execution environment;

performing the at least one task for remote imaging specified; and

while executing an encapsulation, generating another encapsulation,

wherein the at least one specification is an encapsulation, which encapsulates parameters resolved at the first system at execution time by the preboot execution language interpreter from the second system, and

wherein the language agent is based on an encapsulated object-oriented polyphase language.

h. Claim 38: replace claim 38 with the following:

38. A method for remote booting in a preboot execution environment, comprising:

receiving by a first system from a second system a language agent with a preboot execution language interpreter;

generating one or more encapsulations by the language agent;

interpreting one or more encapsulations by the language agent;

receiving by the first system from the second system at least one specification for performing at least one task for remote booting in the preboot execution environment;

interpreting by the language agent the at least one specification for performing at least one task for remote booting in the preboot execution environment;

performing the at least one task for remote booting specified; and

while executing an encapsulation, generating another encapsulation,

wherein the at least one specification is an encapsulation, which encapsulates parameters resolved at the first system at execution time by the preboot execution language interpreter from the second system, and

wherein the language agent is based on an encapsulated object-oriented polyphase language.

i. Claim 43: replace claim 43 with the following:

43. A system for specifying computing tasks in a preboot execution environment, comprising:

a computer readable medium comprising:

a language agent with a preboot execution specification generator; and

a definition for at least one specification for performing at least one computing task in a preboot execution environment,

wherein the at least one specification is an encapsulation, encapsulating parameters to be resolved at a second system at execution time by a preboot execution language interpreter located at the second system,

wherein the language agent is configured to generate one or more encapsulations and to interpret one or more encapsulations,

wherein the language agent is based on an encapsulated object-oriented polyphase language, and

wherein while executing an encapsulation, the language agent is configured to generate another encapsulation.

j. Claim 48: replace claim 48 with the following:

48. A system for specifying tasks for image installation in a preboot execution environment, comprising:

a computer readable medium comprising:

a language agent with a preboot execution specification generator; and

a definition for at least one specification for performing at least one task for image installation in a preboot execution environment,

wherein the at least one specification is an encapsulation, encapsulating parameters to be resolved at a second system at execution time by a preboot execution language interpreter located at the second system,

wherein the language agent is configured to generate one or more encapsulations and to interpret one or more encapsulations,

wherein the language agent is based on an encapsulated object-oriented polyphase language, and

wherein while executing an encapsulation, the language agent is configured to generate another encapsulation.

k. Claim 54: replace claim 54 with the following:

54. A system for specifying remote imaging tasks in a preboot execution environment, comprising:

a computer readable medium comprising:

a language agent with a preboot execution specification generator; and

a definition for at least one specification for performing at least one task for remote imaging in a preboot execution environment,

wherein the at least one specification is an encapsulation, encapsulating parameters to be resolved at a second system at execution time by a preboot execution language interpreter located at the second system,

wherein the language agent is configured to generate one or more encapsulations and to interpret one or more encapsulations,

wherein the language agent is based on an encapsulated object-oriented polyphase language, and

wherein while executing an encapsulation, the language agent is configured to generate another encapsulation.

l. Claim 60: replace claim 60 with the following:

60. A system for specifying remote booting tasks in a preboot execution environment, comprising:

a computer readable medium comprising:

a language agent with a preboot execution specification generator; and

a definition for at least one specification for performing at least one task for remote booting in a preboot execution environment,

wherein the at least one specification is an encapsulation, encapsulating parameters to be resolved at a second system at execution time by a preboot execution language interpreter located at the second system,

wherein the language agent is configured to generate one or more encapsulations and to interpret one or more encapsulations,

wherein the language agent is based on an encapsulated object-oriented polyphase language, and

wherein while executing an encapsulation, the language agent is configured to generate another encapsulation.

m. Claim 66: replace claim 66 with the following:

66. A method for specifying computing tasks in a preboot execution environment, comprising:

providing at a first system a language agent with a preboot execution specification generator;

providing at the first system at least one definition for at least one computing task in a preboot execution environment;

generating at the first system a preboot execution specification from the at least one definition utilizing the language agent with a preboot execution specification generator;

generating one or more encapsulations by the language agent;

interpreting one or more encapsulations by the language agent; and

while executing an encapsulation, generating another encapsulation,

wherein the preboot execution specification is an encapsulation, encapsulating parameters to be resolved at a second system at execution time by a preboot execution language interpreter located at the second system, and

wherein the language agent is based on an encapsulated object-oriented polyphase language.

n. Claim 70: replace claim 70 with the following:

70. A method for specifying computing tasks for image installation in a preboot execution environment, comprising:

providing at a first system a language agent with a preboot execution specification generator;

providing at the first system at least one definition for at least one computing task for image installation in a preboot execution environment;

generating at the first system a preboot execution specification from the at least one definition utilizing the language agent with a preboot execution specification generator;

generating one or more encapsulations by the language agent;

interpreting one or more encapsulations by the language agent; and

while executing an encapsulation, generating another encapsulation,

wherein the preboot execution specification is an encapsulation, encapsulating parameters to be resolved at a second system at execution time by a preboot execution language interpreter located at the second system, and

wherein the language agent is based on an encapsulated object-oriented polyphase language.

o. Claim 75: replace claim 75 with the following:

75. A method for specifying remote imaging in a preboot execution environment, comprising:

providing at a first system a language agent with a preboot execution specification generator;

providing at the first system at least one definition for at least one computing task for remote imaging in a preboot execution environment;

generating at the first system a preboot execution specification from the at least one definition utilizing the language agent with a preboot execution specification generator;

generating one or more encapsulations by the language agent;

interpreting one or more encapsulations by the language agent; and

while executing an encapsulation, generating another encapsulation,

wherein the preboot execution specification is an encapsulation, encapsulating parameters to be resolved at a second system at execution time by a preboot execution language interpreter located at the second system, and

wherein the language agent is based on an encapsulated object-oriented polyphase language.

p. Claim 80: replace claim 80 with the following:

80. A method for specifying remote booting operations in a preboot execution environment, comprising:

providing at a first system a language agent with a preboot execution specification generator;

providing at the first system at least one definition for at least one computing task for remote booting in a preboot execution environment;

generating at the first system a preboot execution specification from the at least one definition utilizing the language agent with a preboot execution specification generator;

generating one or more encapsulations by the language agent;

interpreting one or more encapsulations by the language agent; and

while executing an encapsulation, generating another encapsulation,

wherein the preboot execution specification is an encapsulation, encapsulating parameters to be resolved at a second system at execution time by a preboot execution language interpreter located at the second system, and

wherein the language agent is based on an encapsulated object-oriented polyphase language.

q. Claim 85: replace claim 85 with the following:

85. A system for encapsulated platform imaging, comprising:

a computer readable medium comprising:

a language agent with an encapsulated language interpreter for executing an encapsulation, the language agent from a second system, the encapsulation from the second system,

wherein the encapsulation contains all instructions and data necessary to install an operating system onto a computing device,

wherein the encapsulation encapsulates parameters to be resolved at the system at execution time by the encapsulated language interpreter from the second system,

wherein the language agent is configured to generate one or more encapsulations and to interpret one or more encapsulations,

wherein the language agent is based on an encapsulated object-oriented polyphase language, and

wherein while executing an encapsulation, the language agent is configured to generate another encapsulation.

r. Claim 92: replace claim 92 with the following:

92. A method for encapsulated platform imaging, comprising:

receiving by a computing device from another system an encapsulation which contains all instructions and data necessary to install an operating system onto the computing device;

receiving by the computing device from the another system a language agent with an encapsulated language interpreter for executing the encapsulation;

executing by the language agent at the computing device the encapsulation to install the operating system onto the computing device;

generating one or more encapsulations by the language agent;

interpreting one or more encapsulations by the language agent; and

while executing an encapsulation, generating another encapsulation,

wherein the encapsulation encapsulates parameters resolved by the language agent at the computing device at execution time, the language agent located at the computing device, and

wherein the language agent is based on an encapsulated object-oriented polyphase language.

s. Claim 101: replace claim 101 with the following:

101. A system for encapsulated platform imaging, comprising:

a computer readable medium comprising:

a language agent with an encapsulation generator for defining and creating an encapsulation,

wherein the encapsulation contains all instructions and data necessary to install an operating system onto a computing device,

wherein the encapsulation encapsulates parameters to be resolved at a second system at execution time by a language interpreter located at the second system,

wherein the language agent is configured to generate one or more encapsulations and to interpret one or more encapsulations,

wherein the language agent is based on an encapsulated object-oriented polyphase language, and

wherein while executing an encapsulation, the language agent is configured to generate another encapsulation.

t. Claim 102: replace claim 102 with the following:

102. A method for encapsulated platform imaging, comprising:

providing at a first system a language agent with an encapsulation generator;

providing at the first system a definition for an encapsulation;

generating at the first system from the definition an encapsulation containing all instructions and data necessary to install an operating system onto a computing device by utilizing the language agent with an encapsulation generator;

generating one or more encapsulations by the language agent;

interpreting one or more encapsulations by the language agent; and

while executing an encapsulation, generating another encapsulation,

wherein the encapsulation encapsulates parameters to be resolved at a second system at execution time by a language interpreter located at the second system,

wherein the language agent is based on an encapsulated object-oriented polyphase language.

u. Claim 103: replace claim 103 with the following:

103. A system for executing computing tasks in a preboot execution environment, comprising:

means for receiving by the system a language agent with a preboot execution language interpreter, the language agent originated from a second system;

means for receiving by the system at least one specification for performing at least one computing task in the preboot execution environment, the at least one specification originated from the second system;

means for performing the at least one computing task specified;

means for generating one or more encapsulations;

means for interpreting one or more encapsulations; and

means for, while executing an encapsulation, generating another encapsulation,

wherein the at least one specification is an encapsulation, encapsulating parameters resolved at the system at execution time by the preboot execution language interpreter from the second system,

wherein the language agent is configured to interpret the at least one specification for performing at least one computing task in the preboot execution environment,

wherein the language agent is based on an encapsulated object-oriented polyphase language.

v. Claim 104: replace claim 104 with the following:

104. A system for specifying computing tasks in a preboot execution environment, comprising:

means for providing at the system a language agent with a preboot execution specification generator;

means for providing at the system at least one definition for at least one computing task in a preboot execution environment;

means for generating at the system a preboot execution specification from the at least one definition utilizing the language agent with a preboot execution specification generator;

means for generating one or more encapsulations;

means for interpreting one or more encapsulations; and

means for, while executing an encapsulation, generating another encapsulation,

wherein the preboot execution specification is an encapsulation, encapsulating parameters to be resolved at a second system at execution time by a preboot execution language interpreter located at the second system, and

wherein the language agent is based on an encapsulated object-oriented polyphase language.

Reasons for Allowance

4. The following is an examiner's statement of reasons for allowance: the prior art of record fails to teach or suggest the claimed invention. Specifically, the prior art of record fails to teach or suggest a language agent based on an encapsulated object-oriented polyphase language, the language agent, while executing an encapsulation, generating another encapsulation.

5. The prior art of record, Paul and Murphy teaches a language agent with a preboot execution language interpreter, the language agent from a second system; at least one specification for performing at least one computing task in a preboot execution environment, the at least one specification from the second system, wherein the language agent is configured to interpret the at least one specification for perform at least one computing task in the preboot execution environment, and configured to perform the at least one computing task specified, wherein the at least one specification from the second system is an encapsulation, encapsulating parameters to be resolved at the system at execution time by the preboot execution language interpreter from the second system. However, Paul and Murphy fail to teach the independent claims as recited where the language agent is based on an encapsulated object-oriented polyphase language, and the language agent, while executing an encapsulation, generates another encapsulation.

6. Any comment considered necessary by applicant must be submitted no later than the payment of the issue fee and, to avoid processing delays, should preferably accompany the issue

fee. Such submissions should be clearly labeled "Comments on Statement of Reasons for Allowance."

Conclusion

7. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Jue S. Wang whose telephone number is (571) 270-1655. The examiner can normally be reached on M-Th 7:30 am - 5:00pm (EST).

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Lewis Bullock can be reached on 571-272-3759. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Lewis A. Bullock, Jr./
Supervisory Patent Examiner, Art Unit 2193

/Jue S Wang/
Examiner, Art Unit 2193